Wavelets and Approximation

Ronald DeVore

University of South Carolina

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
07 JAN 2005	**N/A**	**-**

4. TITLE AND SUBTITLE	
Wavelets and Approximation	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of South Carolina**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM001750, Wavelets and Multifractal Analysis (WAMA) Workshop held on 19-31 July 2004., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**135**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Univariate Approximation

- $f(x)$, $x \in [0, 1]$

Univariate Approximation

- $f(x)$, $x \in [0, 1]$
- Approximate f by piecewise constants

Univariate Approximation

- $f(x)$, $x \in [0,1]$
- Approximate f by piecewise constants
- Types of approximation: Linear, Nonlinear

Univariate Approximation

- $f(x)$, $x \in [0,1]$
- Approximate f by piecewise constants
- Types of approximation: Linear, Nonlinear
- Select a metric/norm $\|\cdot\|$ to measure distortion

$$\|f\|_{L_p(\Omega)} := \left(\int_\Omega |f(x)|^p \, dx\right)^{1/p}, \quad 0 < p < \infty$$

Univariate Approximation

- $f(x)$, $x \in [0, 1]$
- Approximate f by piecewise constants
- Types of approximation: Linear, Nonlinear
- Select a metric/norm $\|\cdot\|$ to measure distortion

$$\|f\|_{L_p(\Omega)} := \left(\int_\Omega |f(x)|^p \, dx\right)^{1/p}, \quad 0 < p < \infty$$

- $\|f\|_{L_\infty(\Omega)} := \sup_{x \in \Omega} |f(x)|, \quad p = \infty$

Linear approximation

- Divide $[0, 1]$ into n equal length intervals

Linear approximation

- Divide $[0,1]$ into n equal length intervals
- $\Pi_n := \{[\frac{k}{n}, \frac{k+1}{n}), \quad k = 0, \ldots, n-1\}$

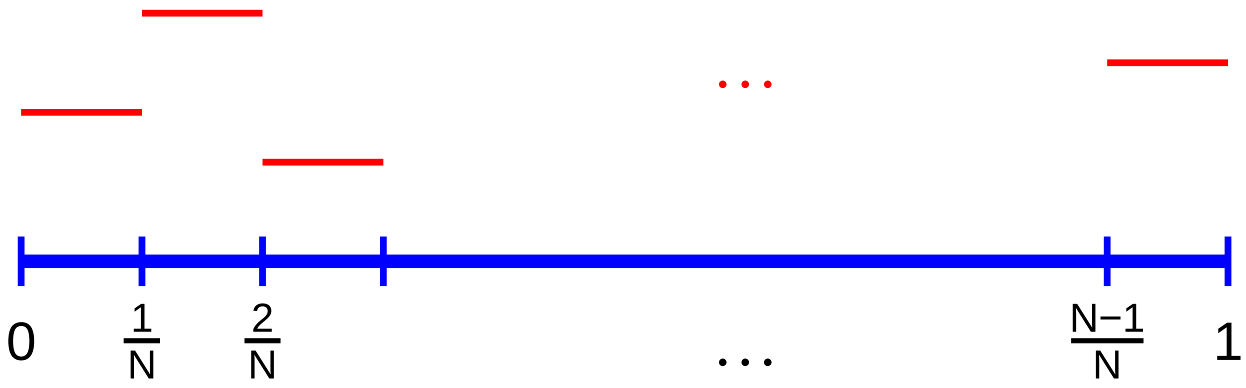
Linear approximation

- Divide $[0,1]$ into n equal length intervals
- $\Pi_n := \{[\frac{k}{n}, \frac{k+1}{n}), \quad k = 0, \ldots, n-1\}$
- $\mathcal{S}_n := \{S : S \; is \; constant \; on \; each \; I \in \Pi_n\}$

Linear approximation

- Divide $[0,1]$ into n equal length intervals
- $\Pi_n := \{[\frac{k}{n}, \frac{k+1}{n}), \quad k = 0, \ldots, n-1\}$
- $\mathcal{S}_n := \{S : S \; is \; constant \; on \; each \; I \in \Pi_n\}$
- $\mathcal{S}_n$ is a linear space

Typical function in $\mathcal{S}_n$



Linear approximation

- Divide $[0,1]$ into n equal length intervals
- $\Pi_n := \{[\frac{k}{n}, \frac{k+1}{n}), \quad k = 0, \dots, n-1\}$
- $\mathcal{S}_n := \{S : S \textit{ is constant on each } I \in \Pi_n\}$
- $\mathcal{S}_n$ is a linear space
- Given $f \in L_p[0,1]$, define error

$$E_n(f)_p := \inf_{S \in \mathcal{S}_n} \|f - S\|_{L_p}$$

Nonlinear: Free Knot Splines

- Divide $[0,1]$ into n intervals $I_i = [\xi_i, \xi_{i+1}]$, $0 = \xi_0 < \xi_1 < \cdots < \xi_n = 1$

Nonlinear: Free Knot Splines

- Divide $[0,1]$ into n intervals $I_i = [\xi_i, \xi_{i+1}]$, $0 = \xi_0 < \xi_1 < \cdots < \xi_n = 1$
- $\Sigma_n := \{S : S \textit{ is constant on each } I_i, i = 1, \ldots, n\}$

Nonlinear: Free Knot Splines

- Divide $[0, 1]$ into n intervals $I_i = [\xi_i, \xi_{i+1}]$,
  $0 = \xi_0 < \xi_1 < \cdots < \xi_n = 1$
- $\Sigma_n := \{S : S \textit{ is constant on each } I_i, i = 1, \ldots, n\}$
- Σ_n is not a linear space: $S_1, S_2 \in \Sigma_n$ but $S_1 + S_2 \in \Sigma_{2n}$

Nonlinear: Free Knot Splines

- Divide $[0, 1]$ into n intervals $I_i = [\xi_i, \xi_{i+1}]$,
  $0 = \xi_0 < \xi_1 < \cdots < \xi_n = 1$
- $\Sigma_n := \{S : S \textit{ is constant on each } I_i, i = 1, \ldots, n\}$
- Σ_n is not a linear space: $S_1, S_2 \in \Sigma_n$ but $S_1 + S_2 \in \Sigma_{2n}$
- Given $f \in L_p[0, 1]$, define error

$$\sigma_n(f)_p := \inf_{S \in \Sigma_n} \|f - S\|_{L_p}$$

Nonlinear: Adaptive

- For each interval I, $E(I)$ the local L_p error in approximating f by constants

$$E(I) := \inf_c \|f - c\|_{L_p(I)}$$

Nonlinear: Adaptive

- For each interval I, $E(I)$ the local L_p error in approximating f by constants

$$E(I) := \inf_c \|f - c\|_{L_p(I)}$$

- Given error tolerance $\epsilon > 0$ generate partition $\mathcal{P}_\epsilon$ such that $E(I) \leq \epsilon$ for all $I \in \mathcal{P}_\epsilon$

Nonlinear: Adaptive

- For each interval I, $E(I)$ the local L_p error in approximating f by constants

$$E(I) := \inf_c \|f - c\|_{L_p(I)}$$

- Given error tolerance $\epsilon > 0$ generate partition $\mathcal{P}_\epsilon$ such that $E(I) \leq \epsilon$ for all $I \in \mathcal{P}_\epsilon$
- I is good if $E(I) \leq \epsilon$

Nonlinear: Adaptive

- For each interval I, $E(I)$ the local L_p error in approximating f by constants

$$E(I) := \inf_c \|f - c\|_{L_p(I)}$$

- Given error tolerance $\epsilon > 0$ generate partition $\mathcal{P}_\epsilon$ such that $E(I) \leq \epsilon$ for all $I \in \mathcal{P}_\epsilon$
- I is good if $E(I) \leq \epsilon$
- I is bad if $E(I) > \epsilon$

Nonlinear: Adaptive

- Initially if $E([0,1]) \leq \epsilon$ then algorithm terminates and $\mathcal{P}_\epsilon := \{[0,1]\}$

Nonlinear: Adaptive

- Initially if $E([0,1]) \le \epsilon$ then algorithm terminates and $\mathcal{P}_\epsilon := \{[0,1]\}$
- if $E([0,1]) > \epsilon$ then set $\mathcal{B}_\epsilon := \{[0,1]\}$ and $\mathcal{G}_\epsilon := \emptyset$

Nonlinear: Adaptive

- Initially if $E([0,1]) \leq \epsilon$ then algorithm terminates and $\mathcal{P}_\epsilon := \{[0,1]\}$
- if $E([0,1]) > \epsilon$ then set $\mathcal{B}_\epsilon := \{[0,1]\}$ and $\mathcal{G}_\epsilon := \emptyset$
- Recursion: For each $I \in \mathcal{B}_\epsilon$ put child J of I in $\mathcal{G}_\epsilon$ if it is good, put it in $\mathcal{B}_\epsilon$ if it is bad. Remove I from $\mathcal{B}_\epsilon$

Nonlinear: Adaptive

- Initially if $E([0,1]) \le \epsilon$ then algorithm terminates and $\mathcal{P}_\epsilon := \{[0,1]\}$
- if $E([0,1]) > \epsilon$ then set $\mathcal{B}_\epsilon := \{[0,1]\}$ and $\mathcal{G}_\epsilon := \emptyset$
- Recursion: For each $I \in \mathcal{B}_\epsilon$ put child J of I in $\mathcal{G}_\epsilon$ if it is good, put it in $\mathcal{B}_\epsilon$ if it is bad. Remove I from $\mathcal{B}_\epsilon$
- Stop when $\mathcal{B}_\epsilon = \emptyset$, $\mathcal{P}_\epsilon := \mathcal{G}_\epsilon$, $N_\epsilon := \#(\mathcal{P}_\epsilon)$

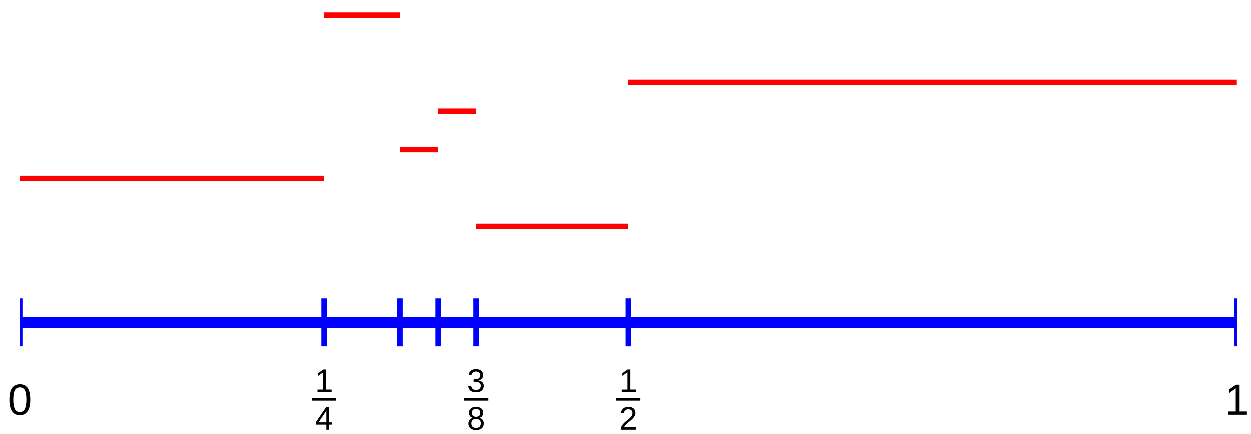
Nonlinear: Adaptive

- $A_{N_\epsilon}(f)$ best approximation to f by piecewise constants on $\mathcal{P}_\epsilon$

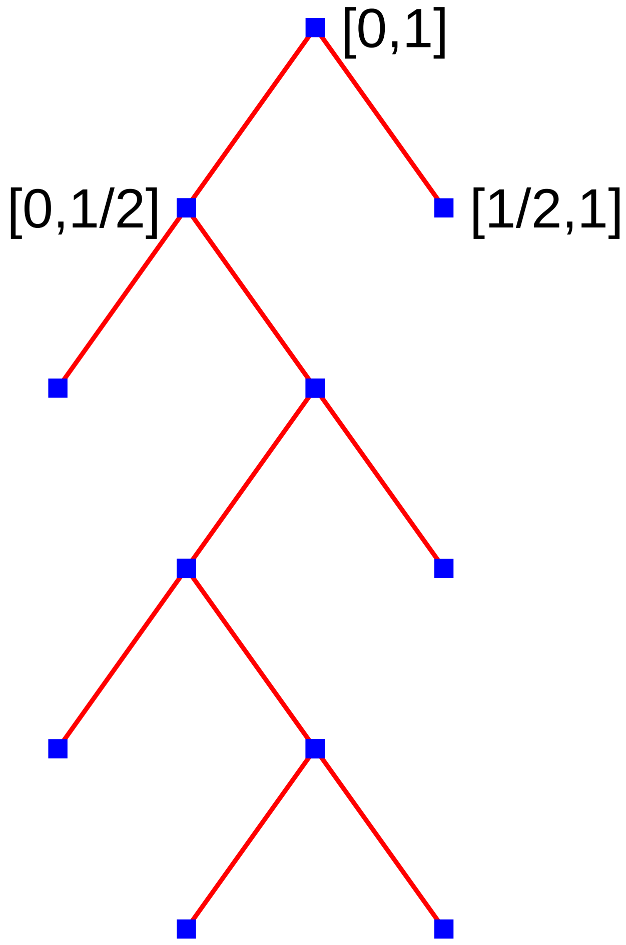
Nonlinear: Adaptive

- $A_{N_\epsilon}(f)$ best approximation to f by piecewise constants on $\mathcal{P}_\epsilon$
- $a_n(f)_p := \inf\{\epsilon : N_\epsilon \le n\}$

Adaptively generated partition



Tree associated to adaptive partition



Comparison

- Approximation classes: $\alpha > 0$
  define $\mathcal{A}^\alpha(L_p,\ linear\ splines)$ as the set of all $f \in L_p[0,1]$ such that
  $$E_n(f)_p \le Cn^{-\alpha}, \quad n \ge 1$$

Comparison

- Approximation classes: $\alpha > 0$
define $\mathcal{A}^\alpha(L_p,\ linear\ splines)$ as the set of all $f \in L_p[0,1]$ such that

$$E_n(f)_p \leq Cn^{-\alpha}, \quad n \geq 1$$

- Similarly define $\mathcal{A}^\alpha(L_p)$ for the other forms of approximation

Comparison

- Approximation classes: $\alpha > 0$ define $\mathcal{A}^\alpha(L_p,\ linear\ splines)$ as the set of all $f \in L_p[0,1]$ such that

$$E_n(f)_p \le C n^{-\alpha}, \quad n \ge 1$$

- Similarly define $\mathcal{A}^\alpha(L_p)$ for the other forms of approximation
- $\mathcal{A}_q^\alpha(L_p)$ finer scaling: same approximation order α

$$|f|_{\mathcal{A}_q^\alpha(L_p)} := (\sum_{n=1}^{\infty}[n^\alpha E_n(f)_p]^q)^{1/q}$$

Approximation Classes: Linear

- Fix the L_p space to measure error

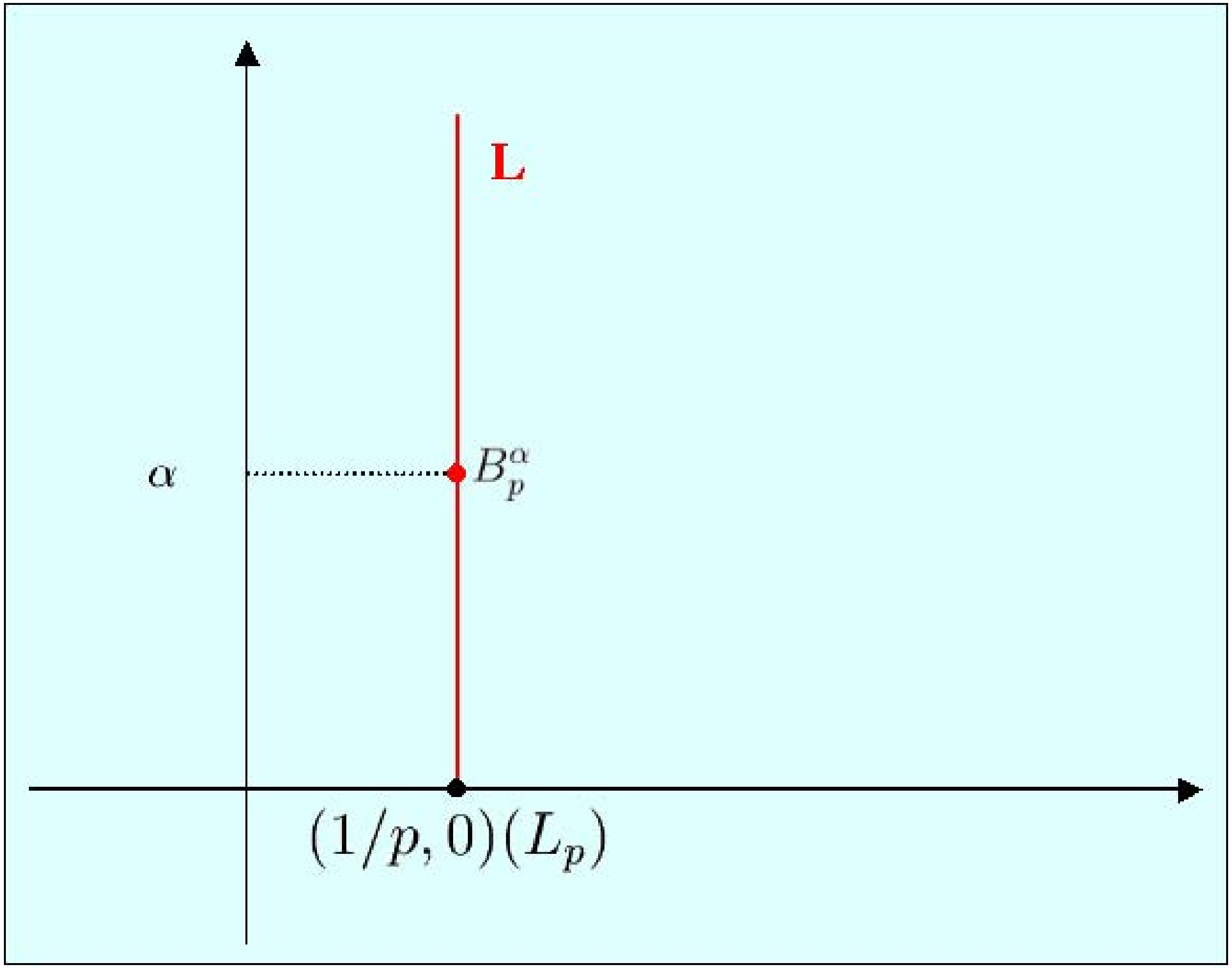
Approximation Classes: Linear

- Fix the L_p space to measure error
- $A^s(L_p,\ linear) = B^s_\infty(L_p)$

Approximation Classes: Linear

- Fix the L_p space to measure error
- $A^s(L_p,\ linear) = B^s_\infty(L_p)$
- Proved by Scherer +

pproximation: $\mathcal{A}^s_\infty(L_p)$ Besov space of smo



Approximation Classes: free knot splines

- Fix the L_p space to measure error

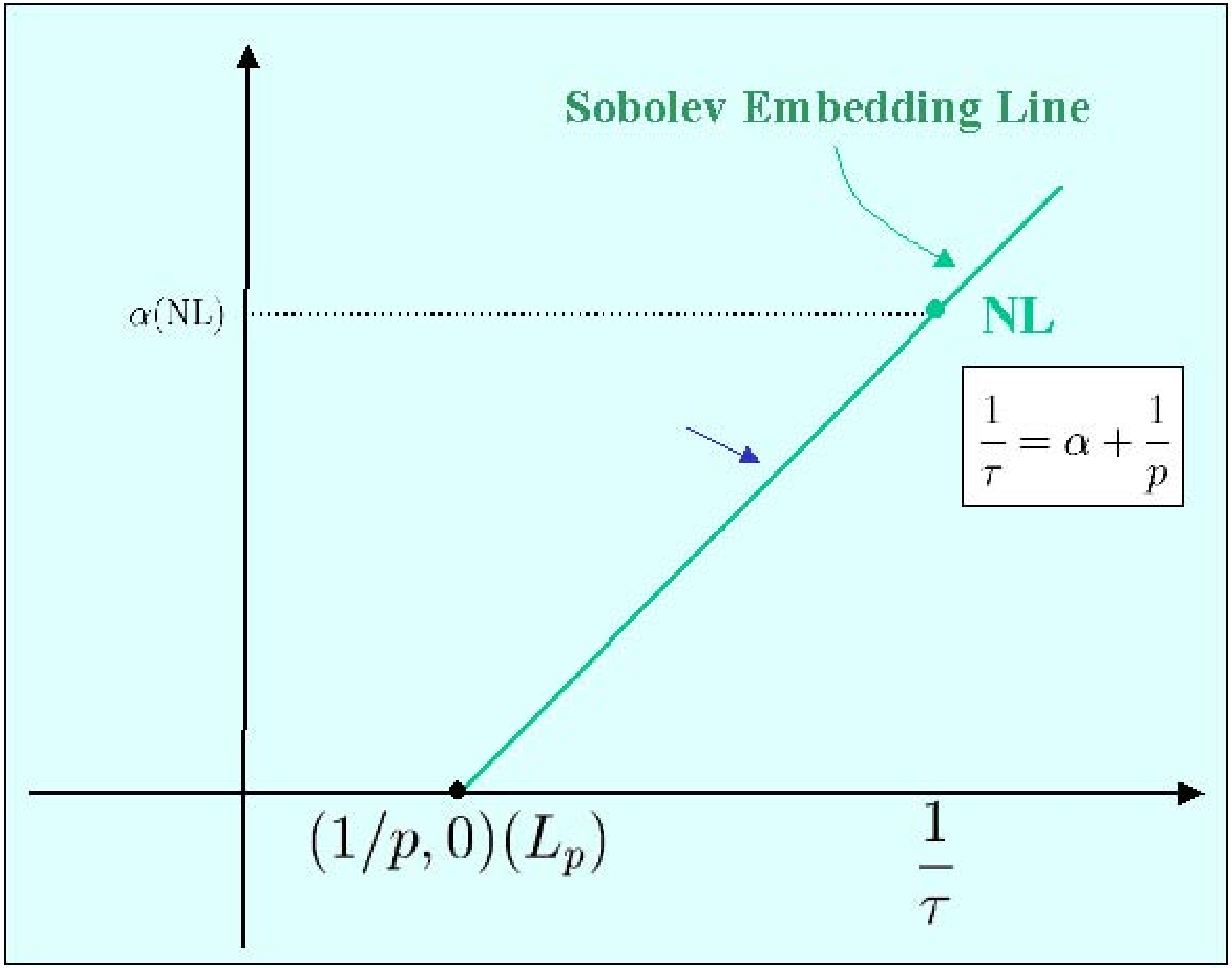
Approximation Classes: free knot splines

- Fix the L_p space to measure error
- $A^s_\tau(L_p,\ nonlinear) = B^s_\tau(L_\tau)$, $\frac{1}{\tau} = s + \frac{1}{p}$

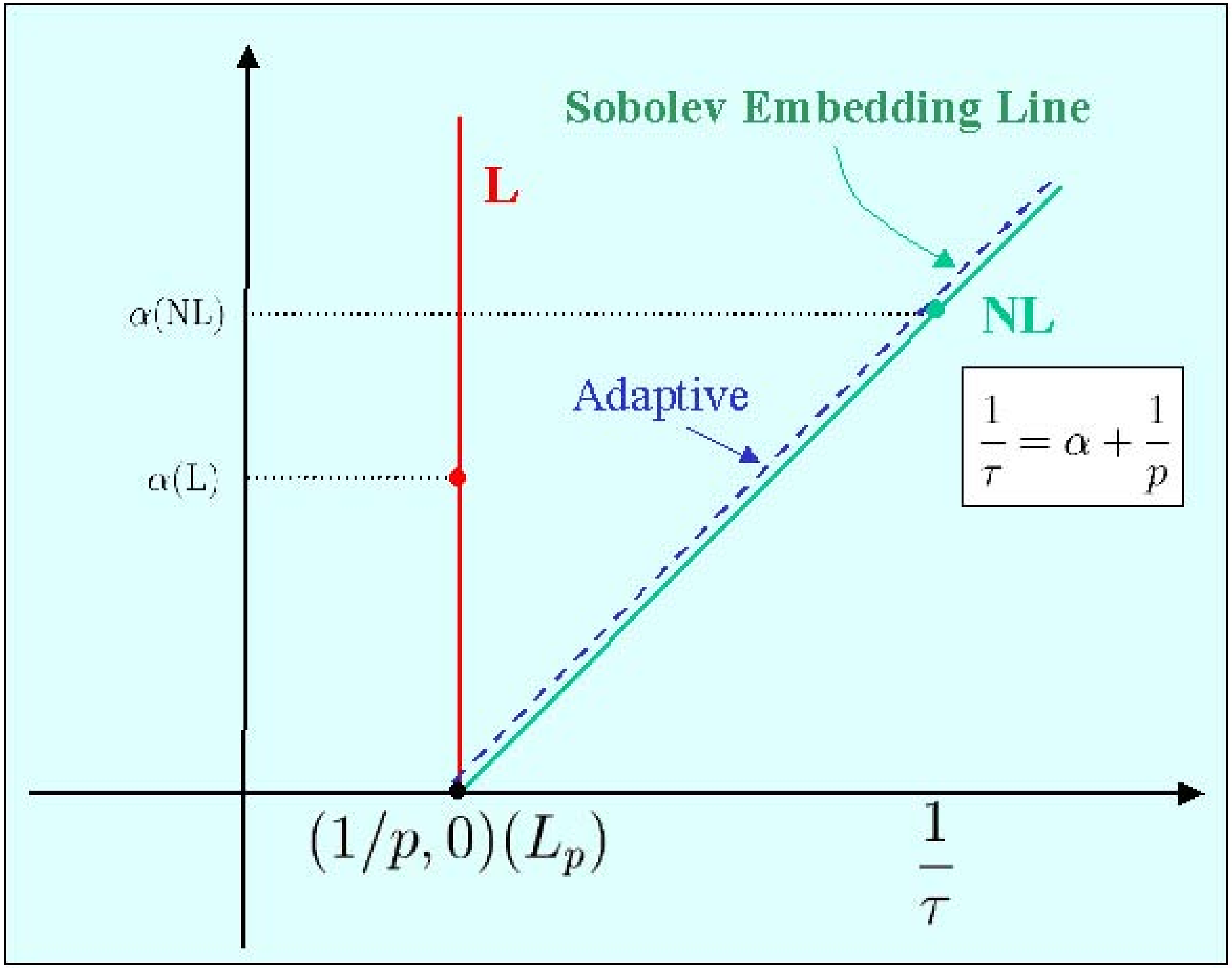
Approximation Classes: free knot splines

- Fix the L_p space to measure error
- $A^s_\tau(L_p,\ nonlinear) = B^s_\tau(L_\tau)$, $\frac{1}{\tau} = s + \frac{1}{p}$
- Petrushev, DeVore-Popov (splines); DeVore-Jawerth-Popov (wavelets)

Approximation class: free knot splines



Adaptive approximation



Example: Approximation in L_∞

- $p = \infty$ approximation order $O(n^{-1})$

Example: Approximation in L_∞

- $p = \infty$ approximation order $O(n^{-1})$
- Linear approximation $f' \in L_\infty$

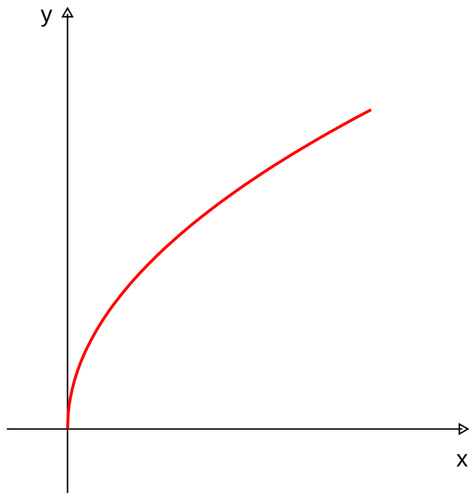
Example: Approximation in L_∞

- $p = \infty$ approximation order $O(n^{-1})$
- Linear approximation $f' \in L_\infty$
- Nonlinear approximation (free knot splines) $f' \in L_1$ or $f \in BV$

Example: Approximation in L_∞

- $p = \infty$ approximation order $O(n^{-1})$
- Linear approximation $f' \in L_\infty$
- Nonlinear approximation (free knot splines) $f' \in L_1$ or $f \in BV$
- Adaptive approximation $f' \in LlogL$: for example $f' \in L_p$ for some $p > 1$

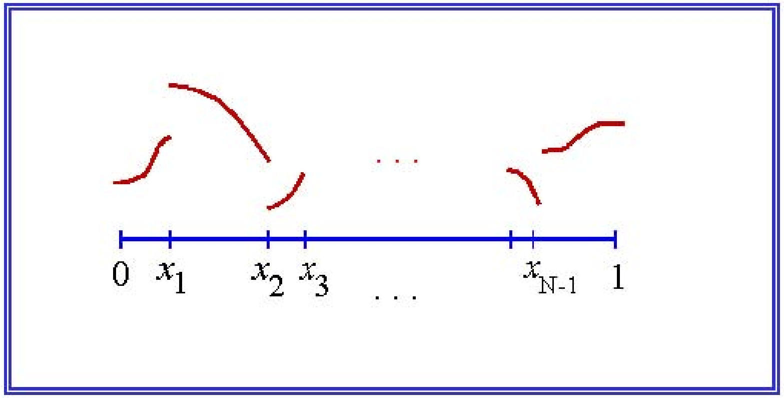
Example: $f(x) = x^\alpha$**,** $0 < \alpha < 1 - 1/p$



$$E_n(f)_p \approx Cn^{-(\alpha+1/p)} \qquad \sigma_n(f)_p \leq Cn^{-1}$$

Break points/ wavelets concentrate near singularity at 0

Example: piecewise smooth

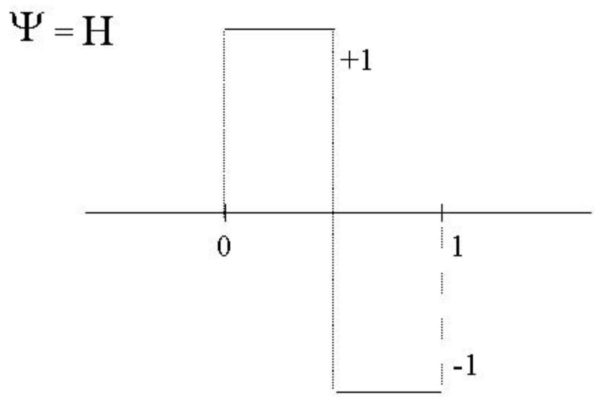


$$E_n(f)_p \geq Cn^{-1/p} \quad \sigma_n(f)_p \leq Cn^{-1}$$

Breakpoints/wavelets concentrate near singularities

Wavelets: Haar Wavelet

$$H(x) := \begin{cases} -1, & x \in [0, 1/2) \\ +1, & x \in [1/2, 1] \end{cases},$$



Wavelets: Haar Basis

- $H_I(x) := 2^{j/2} H(2^j x - k)$, $I = [k2^{-j}, (k+1)2^{-j}]$

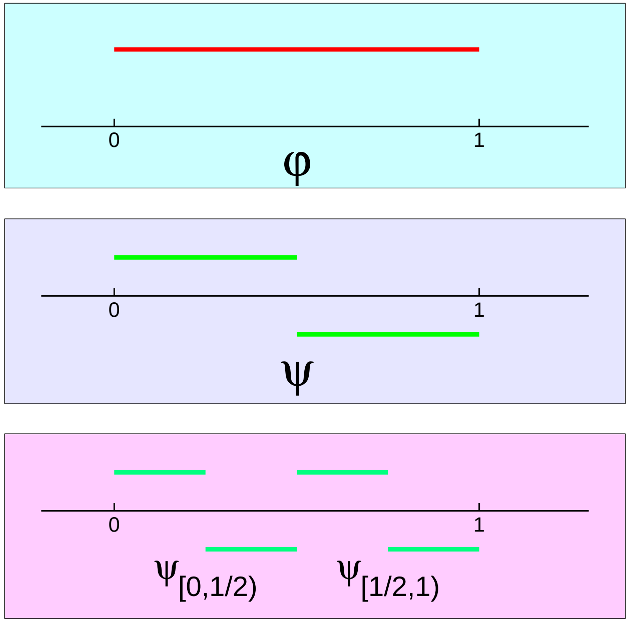
Wavelets: Haar Basis

- $H_I(x) := 2^{j/2}H(2^jx - k)$, $I = [k2^{-j}, (k+1)2^{-j}]$
- $\mathcal{D}_+ := \{I \in \mathcal{D} : |I| \leq 1\}$

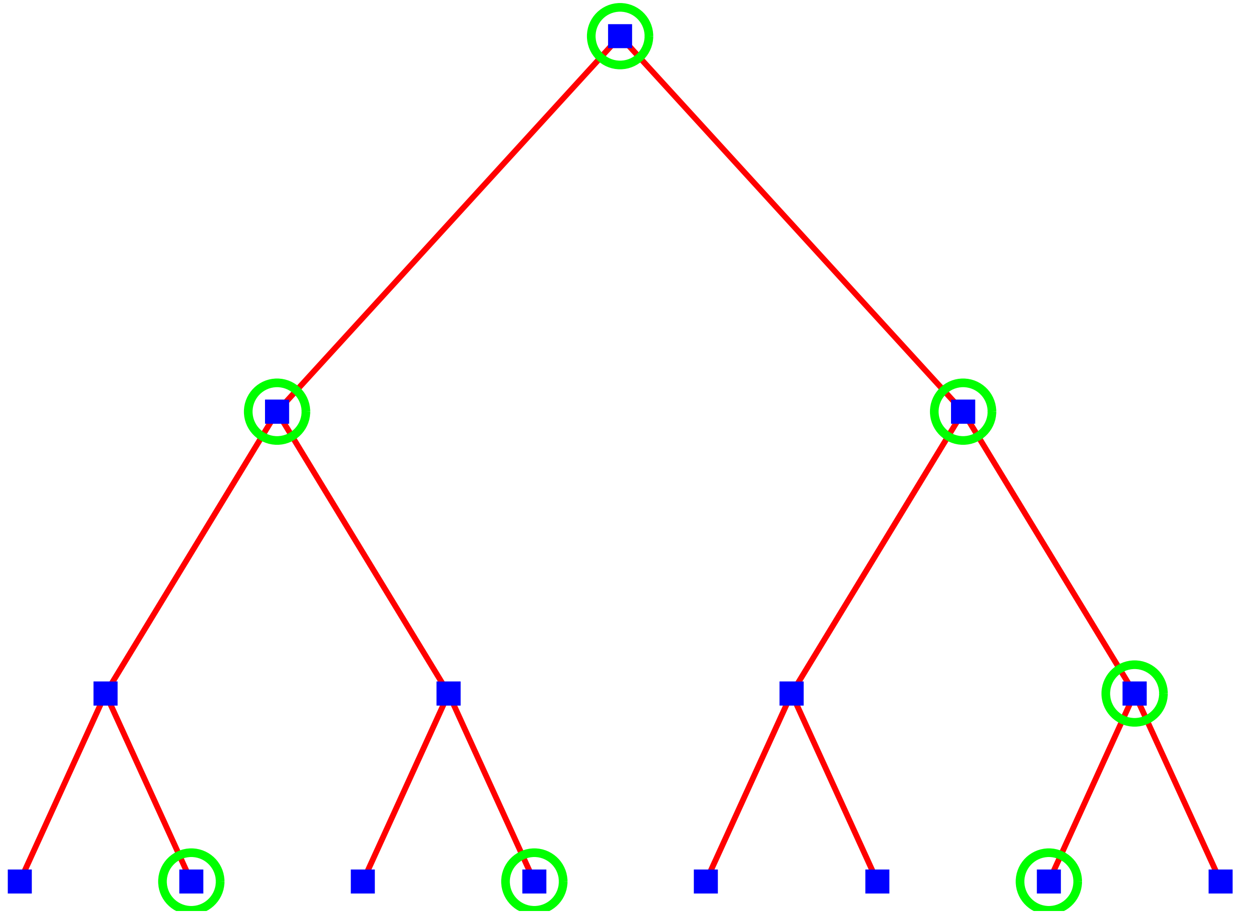
Wavelets: Haar Basis

- $H_I(x) := 2^{j/2}H(2^j x - k)$, $I = [k2^{-j}, (k+1)2^{-j}]$
- $\mathcal{D}_+ := \{I \in \mathcal{D} : |I| \leq 1\}$
- $\{\chi_{[0,1]}\} \cup \{H_I\}_{I \in \mathcal{D}_+}$ is a complete orthonormal system in $L_2[0,1]$

Haar Basis



Wavelet tree



Linear Approximation with Haar Basis

- Natural ordering of dyadic intervals

Linear Approximation with Haar Basis

- Natural ordering of dyadic intervals
- X_n span of first n Haar functions

Linear Approximation with Haar Basis

- Natural ordering of dyadic intervals
- X_n span of first n Haar functions
- X_n is a linear space

Linear Approximation with Haar Basis

- Natural ordering of dyadic intervals
- X_n span of first n Haar functions
- X_n is a linear space
- $E_n^w(f)_p := \inf_{g \in X_n} \|f - g\|_{L_p[0,1]}$

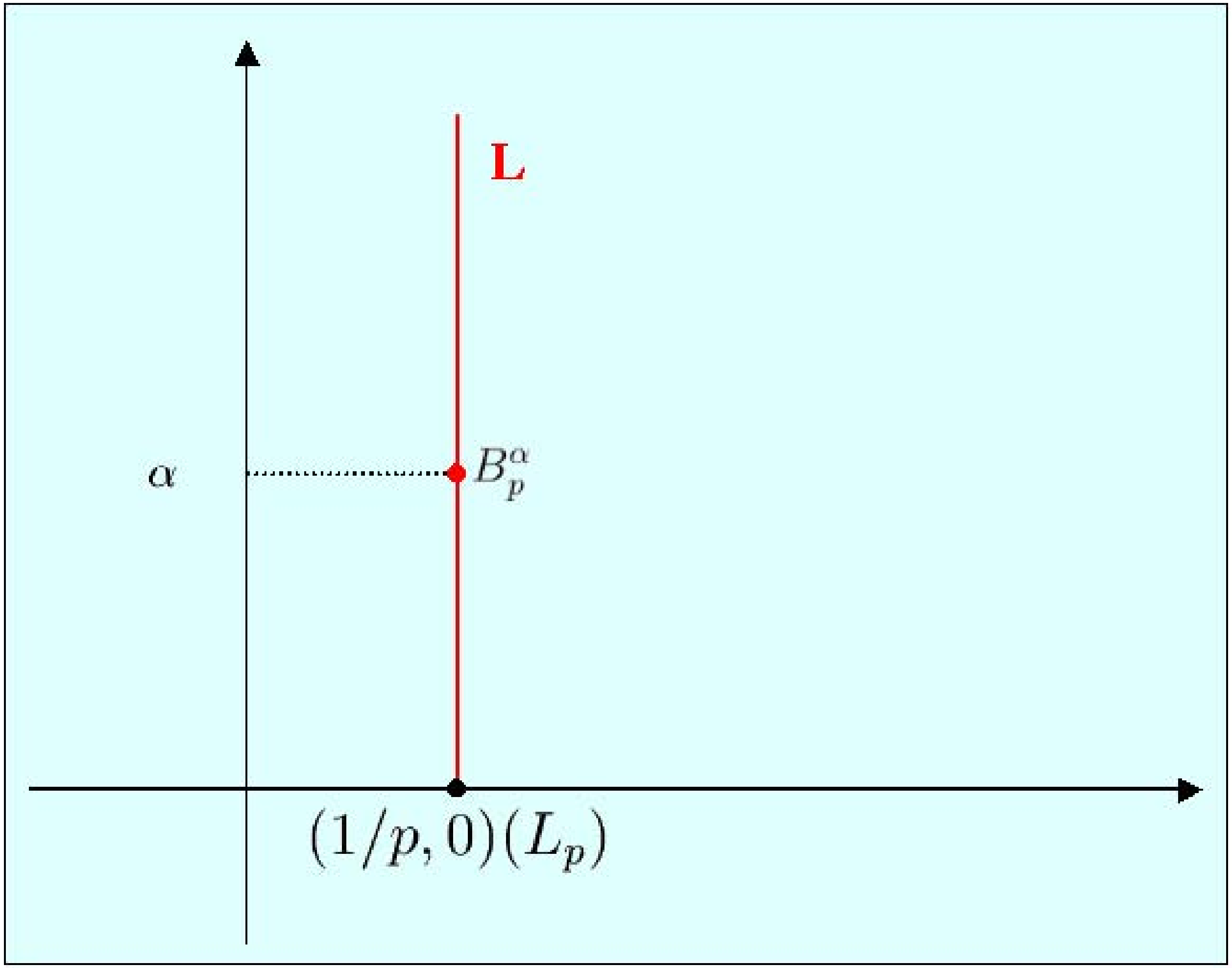
Linear Approximation with Haar Basis

- Natural ordering of dyadic intervals
- X_n span of first n Haar functions
- X_n is a linear space
- $E_n^w(f)_p := \inf_{g \in X_n} \|f - g\|_{L_p[0,1]}$
- This is linear approximation because X_n is a linear space

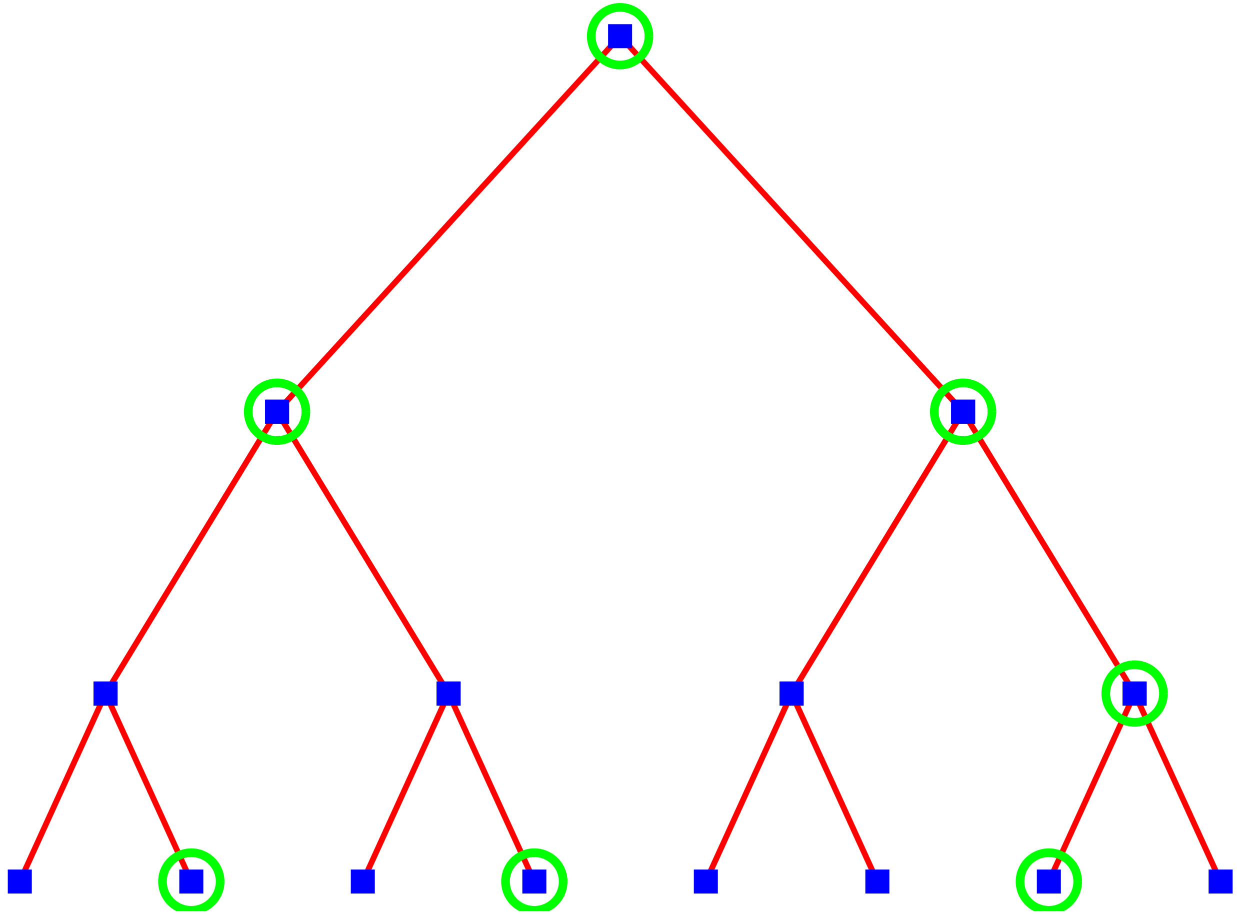
Linear Approximation with Haar Basis

- Natural ordering of dyadic intervals
- X_n span of first n Haar functions
- X_n is a linear space
- $E_n^w(f)_p := \inf_{g \in X_n} \|f - g\|_{L_p[0,1]}$
- This is linear approximation because X_n is a linear space
- The approximation classes for linear approximation with Haar wavelets are identical to those with piecewise constants.

Linear Wavelet: $\mathcal{A}^s_\infty(L_p) = B^s_\infty(L_p)$



n-term approximation



Nonlinear Approximation with Haar Basis

- n-term approximation

Nonlinear Approximation with Haar Basis

- n-term approximation
- $\Sigma_n := \{S = \sum_{I \in \Lambda} c_I H_I : \#(\Lambda) \leq n\}$

Nonlinear Approximation with Haar Basis

- n-term approximation
- $\Sigma_n := \{S = \sum_{I \in \Lambda} c_I H_I : \#(\Lambda) \leq n\}$
- Σ_n is not a linear space

Nonlinear Approximation with Haar Basis

- n-term approximation
- $\Sigma_n := \{S = \sum_{I\in\Lambda} c_I H_I : \#(\Lambda) \le n\}$
- Σ_n is not a linear space
- $\sigma_n^w(f)_p := \inf_{S\in\Sigma_n} \|f - S\|_{L_p[0,1]}$

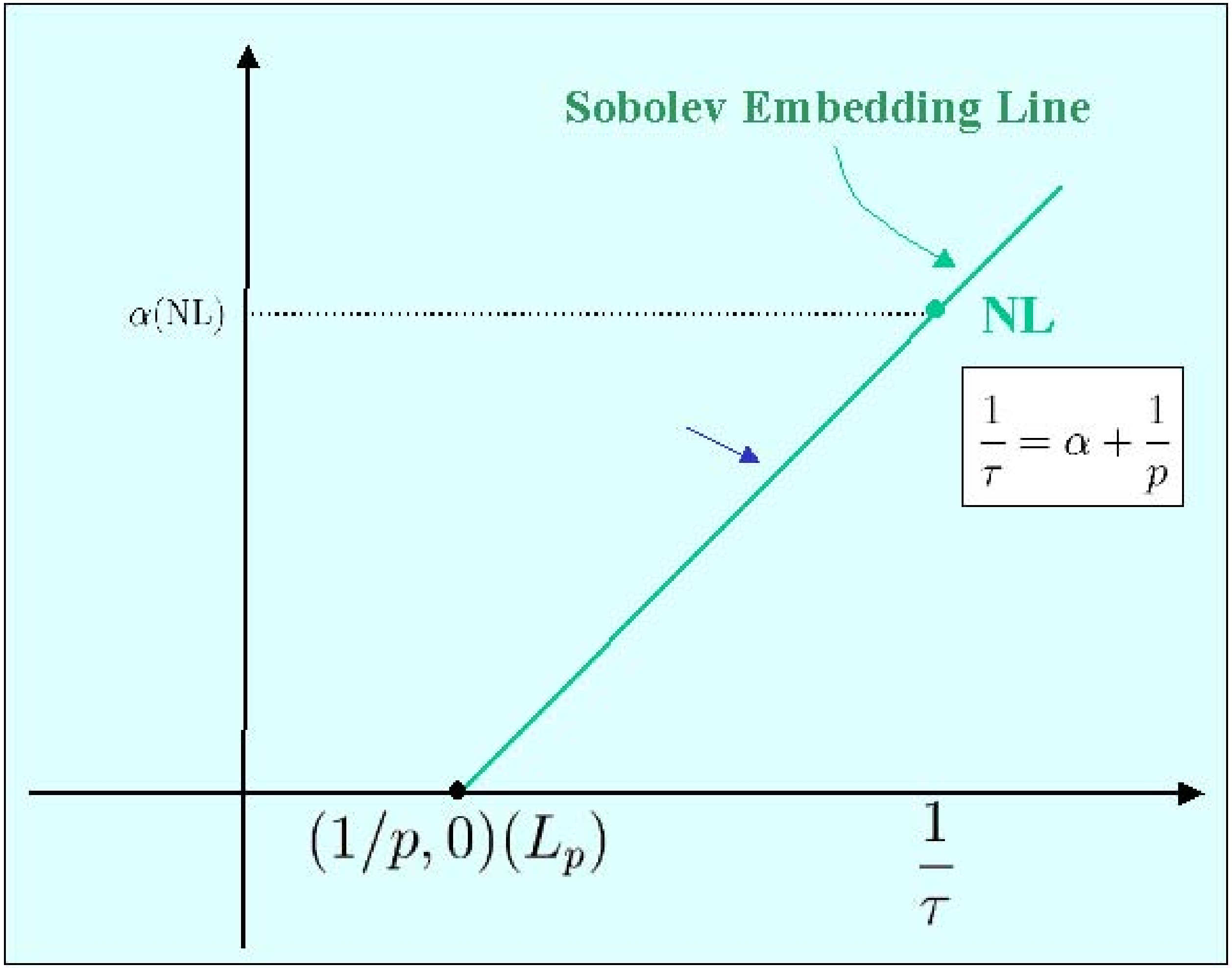
Nonlinear Approximation with Haar Basis

- n-term approximation
- $\Sigma_n := \{S = \sum_{I \in \Lambda} c_I H_I : \#(\Lambda) \leq n\}$
- Σ_n is not a linear space
- $\sigma_n^w(f)_p := \inf_{S \in \Sigma_n} \|f - S\|_{L_p[0,1]}$
- This is nonlinear approximation because decisions are made dependent on f

Nonlinear Approximation with Haar Basis

- n-term approximation
- $\Sigma_n := \{S = \sum_{I \in \Lambda} c_I H_I : \#(\Lambda) \le n\}$
- Σ_n is not a linear space
- $\sigma_n^w(f)_p := \inf_{S \in \Sigma_n} \|f - S\|_{L_p[0,1]}$
- This is nonlinear approximation because decisions are made dependent on f
- The approximation classes for nonlinear approximation with Haar wavelets are identical to those with piecewise constants.

proximation class n-term Haar approxima



So Why Wavelets??

- Simplicity of numerical implementation

So Why Wavelets??

- Simplicity of numerical implementation
- How can we find best n-term approximation??

So Why Wavelets??

- Simplicity of numerical implementation
- How can we find best n-term approximation??
- In L_2 take n terms with largest coefficients

So Why Wavelets??

- Simplicity of numerical implementation
- How can we find best n-term approximation??
- In L_2 take n terms with largest coefficients
- DJP: Same strategy works in L_p, $1 < p < \infty$, and other spaces (Sobolev)

Greedy Approximation in $X = L_p$

- Write $f = \sum_{I \in \mathcal{D}} c_I(f)\psi_I$ with $\|\psi_I\|_X = 1$

Greedy Approximation in $X = L_p$

- Write $f = \sum_{I \in \mathcal{D}} c_I(f) \psi_I$ with $\|\psi_I\|_X = 1$
- $\Lambda_n(f)$ indicies of n largest coefficients $|c_I(f)|$

Greedy Approximation in $X = L_p$

- Write $f = \sum_{I \in \mathcal{D}} c_I(f)\psi_I$ with $\|\psi_I\|_X = 1$
- $\Lambda_n(f)$ indicies of n largest coefficients $|c_I(f)|$
- $G_n(f) := \sum_{I \in \Lambda_n(f)} c_I(f)\psi_I$

Greedy Approximation in $X = L_p$

- Write $f = \sum_{I \in \mathcal{D}} c_I(f)\psi_I$ with $\|\psi_I\|_X = 1$
- $\Lambda_n(f)$ indicies of n largest coefficients $|c_I(f)|$
- $G_n(f) := \sum_{I \in \Lambda_n(f)} c_I(f)\psi_I$
- Temlyakov:

$$\|f - G_n(f)\|_X \leq C_X \sigma_n(f)_X$$

Greedy Approximation in $X = L_p$

- Write $f = \sum_{I \in \mathcal{D}} c_I(f)\psi_I$ with $\|\psi_I\|_X = 1$
- $\Lambda_n(f)$ indicies of n largest coefficients $|c_I(f)|$
- $G_n(f) := \sum_{I \in \Lambda_n(f)} c_I(f)\psi_I$
- Temlyakov:

$$\|f - G_n(f)\|_X \leq C_X \sigma_n(f)_X$$

- Greedy strategy is near optimal

Thresholding is near optimal in L_p

- $\Lambda(f,\eta) := \{I : |c_I(f)| > \eta\}$, $N := \#(\Lambda(f,\eta))$

Thresholding is near optimal in L_p

- $\Lambda(f,\eta) := \{I : |c_I(f)| > \eta\}$, $N := \#(\Lambda(f,\eta))$
- $T_\eta(f) := \sum_{I\in\Lambda(f,\eta)} c_I(f)\psi$

Thresholding is near optimal in L_p

- $\Lambda(f,\eta) := \{I : |c_I(f)| > \eta\}$, $N := \#(\Lambda(f,\eta))$
- $T_\eta(f) := \sum_{I\in\Lambda(f,\eta)} c_I(f)\psi$
- $\|f - T_\eta(f)\|_{L_p} \leq C_p \sigma_N(f)_p$

Generalizations

- X a Banach space: $\{\psi_j\}$ basis

Generalizations

- X a Banach space: $\{\psi_j\}$ basis
- $f = \sum_{j=1}^{\infty} c_j(f)\psi_j$

Generalizations

- X a Banach space: $\{\psi_j\}$ basis
- $f = \sum_{j=1}^{\infty} c_j(f)\psi_j$
- $\Lambda_n(f)$ set of indices of n largest $|c_j(f)|$

Generalizations

- X a Banach space: $\{\psi_j\}$ basis
- $f = \sum_{j=1}^{\infty} c_j(f)\psi_j$
- $\Lambda_n(f)$ set of indices of n largest $|c_j(f)|$
- $G_n(f) := \sum_{j \in \Lambda_n(f)} c_j(f)\psi_j$

Generalizations

- X a Banach space: $\{\psi_j\}$ basis
- $f = \sum_{j=1}^{\infty} c_j(f)\psi_j$
- $\Lambda_n(f)$ set of indices of n largest $|c_j(f)|$
- $G_n(f) := \sum_{j \in \Lambda_n(f)} c_j(f)\psi_j$
- When do we have $\|f - G_n(f)\|_X \leq C_X \sigma_n(f)_X$?

Generalizations

- X a Banach space: $\{\psi_j\}$ basis
- $f = \sum_{j=1}^{\infty} c_j(f)\psi_j$
- $\Lambda_n(f)$ set of indices of n largest $|c_j(f)|$
- $G_n(f) := \sum_{j\in\Lambda_n(f)} c_j(f)\psi_j$
- When do we have $\|f - G_n(f)\|_X \leq C_X \sigma_n(f)_X$?
- Konjagin-Temlyakov: Near optimal equivalent to the basis is unconditional and democratic

Generalizations

- X a Banach space: $\{\psi_j\}$ basis
- $f = \sum_{j=1}^{\infty} c_j(f)\psi_j$
- $\Lambda_n(f)$ set of indices of n largest $|c_j(f)|$
- $G_n(f) := \sum_{j\in\Lambda_n(f)} c_j(f)\psi_j$
- When do we have $\|f - G_n(f)\|_X \le C_X \sigma_n(f)_X$?
- Konjagin-Temlyakov: Near optimal equivalent to the basis is unconditional and democratic
- Democratic

$$\frac{\|\sum_{I\in\Lambda}\psi_I\|_X}{\|\sum_{I\in\Lambda'}\psi_I\|_X} \le C$$

whenever $\#(\Lambda) = \#(\Lambda')$

Wavelet Bases

- Wavelet bases are democratic in L_p, $1 < p < \infty$, in H_p, $p \leq 1$

Wavelet Bases

- Wavelet bases are democratic in L_p, $1 < p < \infty$, in H_p, $p \le 1$
- $\|\sum_{I\in\Lambda}\psi_I\|_{L_p} \approx \#(\Lambda)^{1/p}$

Wavelet Bases

- Wavelet bases are democratic in L_p, $1 < p < \infty$, in H_p, $p \leq 1$
- $\|\sum_{I\in\Lambda} \psi_I\|_{L_p} \approx \#(\Lambda)^{1/p}$
- $$C_1 \min_{j\in\Lambda} |c_j(f)|(\#(\Lambda))^{1/p} \leq \|\sum_{j\in\Lambda} c_j(f)\psi_j\|_{L_p}$$

$$\|\sum_{j\in\Lambda} c_j(f)\psi_j\|_{L_p} \leq C_2 \max_{j\in\Lambda} |c_j(f)|(\#(\Lambda))^{1/p}$$

When do we get $\sigma_n(f)_p = O(n^{-\alpha})$

- ℓ_τ: $\|(c_j)\|_{\ell_\tau} := (\sum_{j=1}^{\infty} |c_j|^\tau)^{1/\tau}$

When do we get $\sigma_n(f)_p = O(n^{-\alpha})$

- ℓ_τ: $\|(c_j)\|_{\ell_\tau} := (\sum_{j=1}^{\infty} |c_j|^\tau)^{1/\tau}$
- rearrangement of (c_j):
  c_n^* the n-th largest of $|c_j|$, $j \in \{1, 2, \dots, \}$

When do we get $\sigma_n(f)_p = O(n^{-\alpha})$

- ℓ_τ: $\|(c_j)\|_{\ell_\tau} := (\sum_{j=1}^{\infty} |c_j|^\tau)^{1/\tau}$
- rearrangement of (c_j):
  c_n^* the n-th largest of $|c_j|$, $j \in \{1, 2, \ldots, \}$
- weak-ℓ_τ :
  $c_n^* \leq M n^{-1/\tau}$

When do we get $\sigma_n(f)_p = O(n^{-\alpha})$

- ℓ_τ: $\|(c_j)\|_{\ell_\tau} := (\sum_{j=1}^{\infty} |c_j|^\tau)^{1/\tau}$
- rearrangement of (c_j):
  c_n^* the n-th largest of $|c_j|$, $j \in \{1, 2, \dots, \}$
- weak-ℓ_τ :
  $c_n^* \leq M n^{-1/\tau}$
- smallest M gives $|(c_j)|_{w\ell_\tau}$

When do we get $\sigma_n(f)_p = O(n^{-\alpha})$

- ℓ_τ: $\|(c_j)\|_{\ell_\tau} := (\sum_{j=1}^{\infty} |c_j|^\tau)^{1/\tau}$
- rearrangement of (c_j):
  c_n^* the n-th largest of $|c_j|$, $j \in \{1, 2, \dots, \}$
- weak-ℓ_τ :
  $c_n^* \leq M n^{-1/\tau}$
- smallest M gives $|(c_j)|_{w\ell_\tau}$
- $X = L_p$, $1 < p < \infty$, $\{\psi_j\}$ greedy basis.
  $(c_j) \in w\ell_\tau,\ 1/\tau = \alpha + 1/p \leftrightarrow$

$$\sigma_n(f)_p, \|f - G_n(f)\|_{L_p} \leq C M n^{-\alpha}, \quad n = 1, 2, \dots$$

When do we get $\sigma_n(f)_p = O(n^{-\alpha})$

- ℓ_τ: $\|(c_j)\|_{\ell_\tau} := (\sum_{j=1}^\infty |c_j|^\tau)^{1/\tau}$
- rearrangement of (c_j):
  c_n^* the n-th largest of $|c_j|$, $j \in \{1, 2, \dots,\}$
- weak-ℓ_τ :
  $c_n^* \le M n^{-1/\tau}$
- smallest M gives $|(c_j)|_{w\ell_\tau}$
- $X = L_p$, $1 < p < \infty$, $\{\psi_j\}$ greedy basis.
  $(c_j) \in w\ell_\tau,\ 1/\tau = \alpha + 1/p \leftrightarrow$

$$\sigma_n(f)_p, \|f - G_n(f)\|_{L_p} \le CMn^{-\alpha}, \quad n = 1, 2, \dots$$

- $\sum_{n=1}^\infty [n^\alpha \|f - G_n(f)\|_{L_p}]^\tau \frac{1}{n} \approx \|(c_j(f)\|_{\ell_\tau}^\tau$

Tree Approximation with Haar Basis

- n-term approximation not numerically implementable: wavelet positions scattered and uncontrolled

Tree Approximation with Haar Basis

- n-term approximation not numerically implementable: wavelet positions scattered and uncontrolled
- Require that the wavelet positions chosen in the approximation lie on a tree with n-nodes

Tree Approximation with Haar Basis

- n-term approximation not numerically implementable: wavelet positions scattered and uncontrolled
- Require that the wavelet positions chosen in the approximation lie on a tree with n-nodes
- $\Sigma_n^t := \{S = \sum_{I \in \Lambda} c_I H_I : \ \Lambda \ a \ tree \ \#(\Lambda) \leq n\}$

Tree Approximation with Haar Basis

- n-term approximation not numerically implementable: wavelet positions scattered and uncontrolled
- Require that the wavelet positions chosen in the approximation lie on a tree with n-nodes
- $\Sigma_n^t := \{S = \sum_{I \in \Lambda} c_I H_I : \ \Lambda \ a \ tree \ \#(\Lambda) \leq n\}$
- $\sigma_n^t(f)_p := \inf_{S \in \Sigma_n^t} \|f - S\|_{L_p[0,1]}$

Tree Approximation with Haar Basis

- n-term approximation not numerically implementable: wavelet positions scattered and uncontrolled
- Require that the wavelet positions chosen in the approximation lie on a tree with n-nodes
- $\Sigma_n^t := \{S = \sum_{I\in\Lambda} c_I H_I : \ \Lambda \ a \ tree \ \#(\Lambda) \le n\}$
- $\sigma_n^t(f)_p := \inf_{S\in\Sigma_n^t} \|f - S\|_{L_p[0,1]}$
- Numerical algorithm $\Lambda_\eta(f)$ as before - complete $\Lambda_\eta(f)$ to a tree Λ_η^t

$$T_\eta^*(f) := \sum_{I\in\Lambda_\eta^t(f)} c_I(f)\psi_I$$

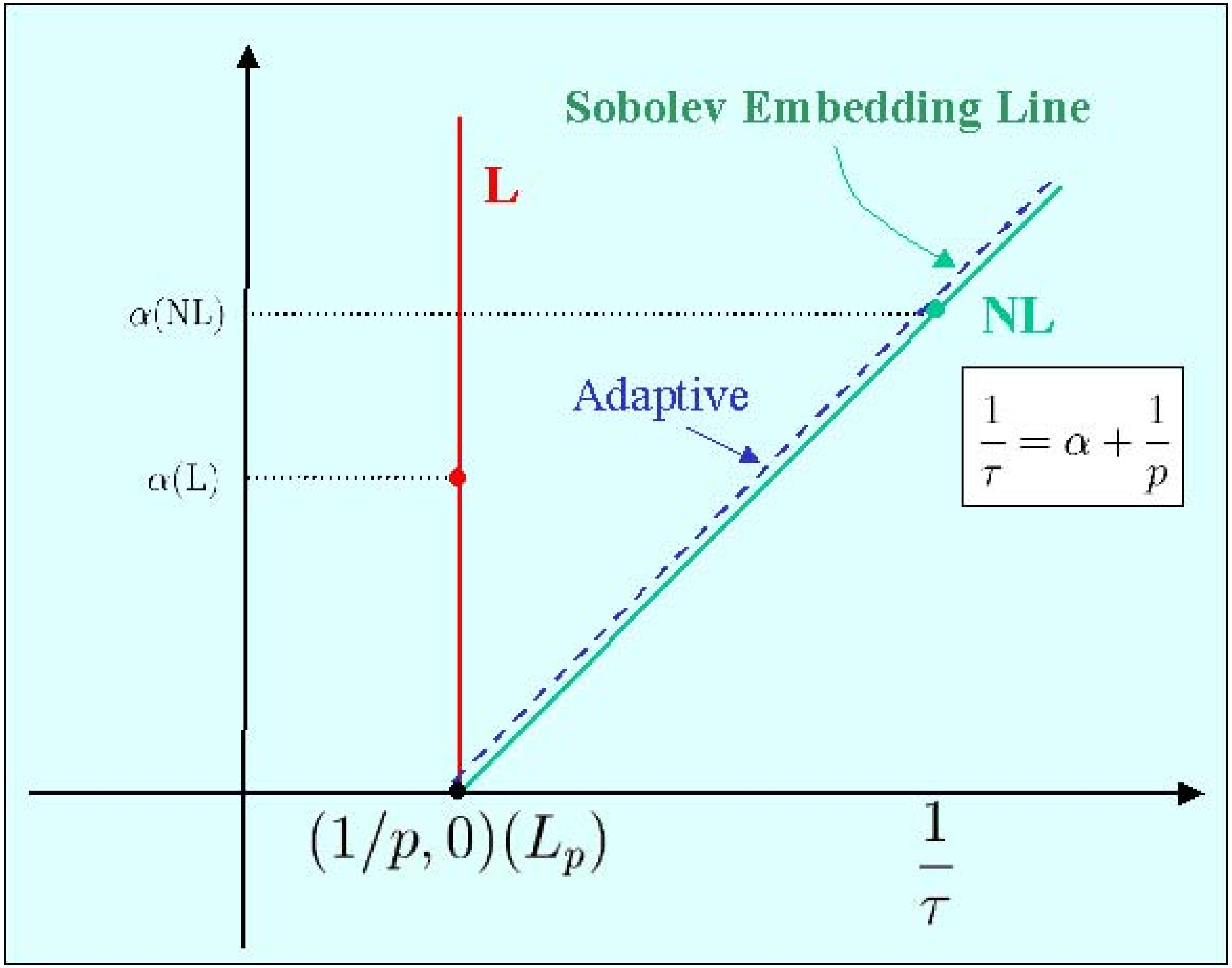
Tree Approximation with Haar Basis

- n-term approximation not numerically implementable: wavelet positions scattered and uncontrolled
- Require that the wavelet positions chosen in the approximation lie on a tree with n-nodes
- $\Sigma_n^t := \{S = \sum_{I\in\Lambda} c_I H_I : \ \Lambda \ a \ tree \ \#(\Lambda) \le n\}$
- $\sigma_n^t(f)_p := \inf_{S\in\Sigma_n^t} \|f - S\|_{L_p[0,1]}$
- Numerical algorithm $\Lambda_\eta(f)$ as before - complete $\Lambda_\eta(f)$ to a tree Λ_η^t

$$T_\eta^*(f) := \sum_{I\in\Lambda_\eta^t(f)} c_I(f)\psi_I$$

- Approximation properties analogous to adaptive approximation

Tree approximation



Extensions

- Can replace Haar wavelets by biorthogonal wavelets

Extensions

- Can replace Haar wavelets by biorthogonal wavelets
- Approximation results now hold provided $\alpha < r$ where ψ has r vanishing moments and smoothness C^r,

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e
- Wavelet Tree Approximation equivalent to adaptive triangulation with isotropic partitioning

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e
- Wavelet Tree Approximation equivalent to adaptive triangulation with isotropic partitioning
- $\mathcal{A}^{\alpha/d}(L_p) = B^{\alpha}_{\infty}(L_p)$

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e
- Wavelet Tree Approximation equivalent to adaptive triangulation with isotropic partitioning
- $\mathcal{A}^{\alpha/d}(L_p) = B^{\alpha}_{\infty}(L_p)$
- Nonlinear Approximation?

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e
- Wavelet Tree Approximation equivalent to adaptive triangulation with isotropic partitioning
- $\mathcal{A}^{\alpha/d}(L_p) = B^{\alpha}_{\infty}(L_p)$
- Nonlinear Approximation?
- No analogue of free knot splines: general triangulation

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e
- Wavelet Tree Approximation equivalent to adaptive triangulation with isotropic partitioning
- $\mathcal{A}^{\alpha/d}(L_p) = B^{\alpha}_{\infty}(L_p)$
- Nonlinear Approximation?
- No analogue of free knot splines: general triangulation
- n-term approximation using multivariate wavelets the same

Multidimensional results

- Multidimensional: Results hold in $\mathbb{R}^d$ with basis ψ_I^e
- Wavelet Tree Approximation equivalent to adaptive triangulation with isotropic partitioning
- $\mathcal{A}^{\alpha/d}(L_p) = B^\alpha_\infty(L_p)$
- Nonlinear Approximation?
- No analogue of free knot splines: general triangulation
- n-term approximation using multivariate wavelets the same
- $\mathcal{A}^{\alpha/d}_\tau(wavelets, L_p) = B^\alpha_\tau(L_p)$, $\frac{1}{\tau} = \frac{\alpha}{d} + \frac{1}{p}$

Compression/Encoding

- How can we evaluate encoders

Compression/Encoding

- How can we evaluate encoders
- Experimental:

  Encoders designed on heuristics

Compression/Encoding

- How can we evaluate encoders
- Experimental:

  Encoders designed on heuristics

- Precise Mathematical Formulation

  Understand rules of game; what it means to be a winner

Compression/Encoding

- How can we evaluate encoders
- Experimental:

  Encoders designed on heuristics

- Precise Mathematical Formulation

  Understand rules of game; what it means to be a winner

- Two essential ingredients

  a. metric ρ to measure distortion

  b. Precise definition of classes K_α to be compressed

Mathematical Formulation

- Distortion: $\rho(S, D_n E_n(S))$

Mathematical Formulation

- Distortion: $\rho(S, D_n E_n(S))$
- Evaluate Performance on a set K of surfaces

$$\delta(K; D_n, E_n) := \sup_{S \in K} \rho(S, D_n E_n(S))$$

Mathematical Formulation

- Distortion: $\rho(S, D_n E_n(S))$
- Evaluate Performance on a set K of surfaces

$$\delta(K; D_n, E_n) := \sup_{S \in K} \rho(S, D_n E_n(S))$$

- Given bit budget n

$$\delta_n(K) := \inf_{E_n, D_n} \delta(K, D_n E_n(S))$$

Mathematical Formulation

- Distortion: $\rho(S, D_n E_n(S))$
- Evaluate Performance on a set K of surfaces

$$\delta(K; D_n, E_n) := \sup_{S \in K} \rho(S, D_n E_n(S))$$

- Given bit budget n

$$\delta_n(K) := \inf_{E_n, D_n} \delta(K, D_n E_n(S))$$

- smallest distortion for the given bit budget

(Near) Optimal Encoding for K

- optimal

$$\delta(K, D_n E_n(S)) = \delta_n(K)$$

(Near) Optimal Encoding for K

- optimal

$$\delta(K, D_n E_n(S)) = \delta_n(K)$$

- near optimal

$$\delta(K, D_n E_n(S)) \leq C\delta_n(K)$$

(Near) Optimal Encoding for K

- optimal

$$\delta(K, D_n E_n(S)) = \delta_n(K)$$

- near optimal

$$\delta(K, D_n E_n(S)) \leq C\delta_n(K)$$

- Typically: $\delta_n(K) \approx n^{-s}$ for some $s > 0$

(Near) Optimal Encoding for K

- optimal

$$\delta(K, D_n E_n(S)) = \delta_n(K)$$

- near optimal

$$\delta(K, D_n E_n(S)) \leq C\delta_n(K)$$

- Typically: $\delta_n(K) \approx n^{-s}$ for some $s > 0$
- Game: Find encoder/decoder E/D: for all values of n and all classes K_α, encoder is near optimal

Description of Optimal Encoding: Kolmo

- Given $\epsilon > 0$

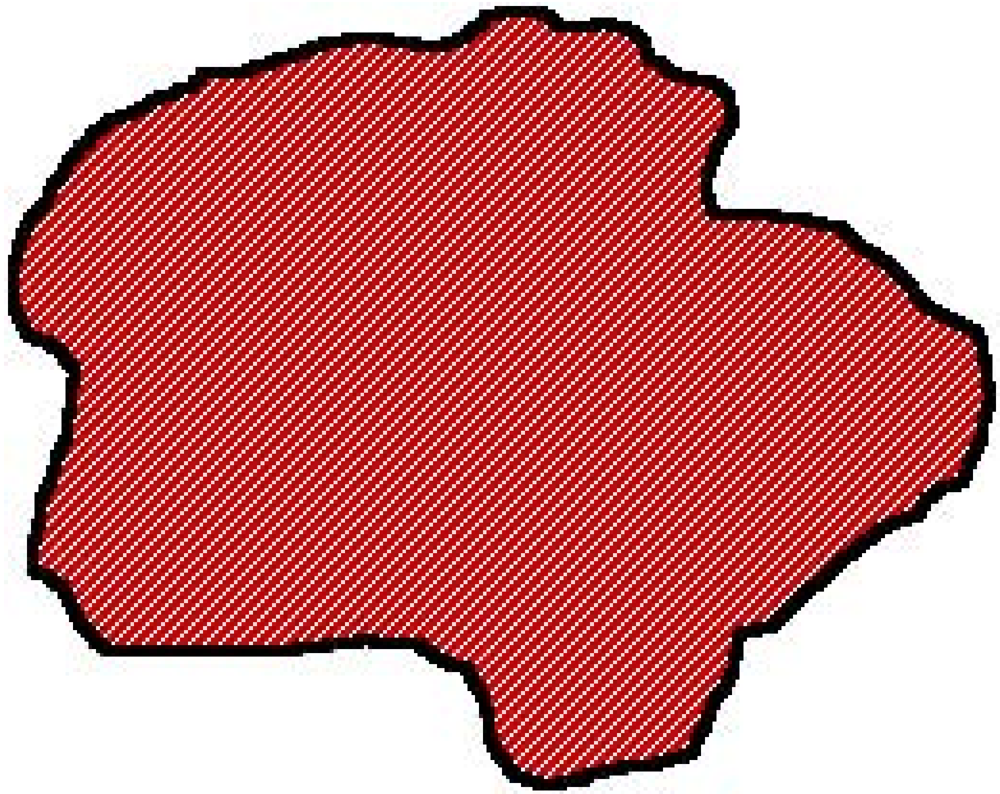
Description of Optimal Encoding: Kolmo

- Given $\epsilon > 0$
- Minimal ϵ cover: $K \subset \cup_{i=1}^{N_\epsilon} \mathcal{B}(S_i, \epsilon)$

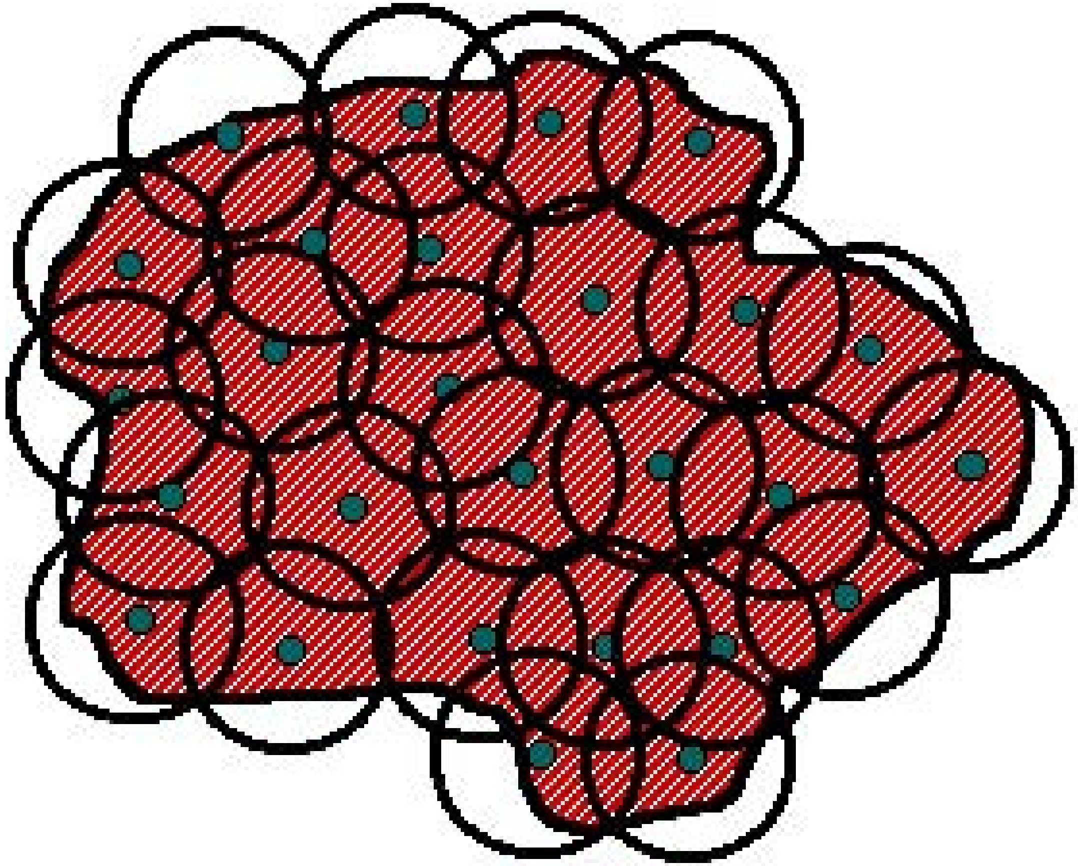
Description of Optimal Encoding: Kolmo

- Given $\epsilon > 0$
- Minimal ϵ cover: $K \subset \cup_{i=1}^{N_\epsilon} \mathcal{B}(S_i, \epsilon)$
- Kolmogorov Entropy $H_\epsilon(K) := \log_2 N_\epsilon(K)$

Covering



Covering



Kolmogorov Entropy

- Given $\epsilon > 0$
- Minimal ϵ cover: $K \subset \cup_{i=1}^{N_\epsilon} \mathcal{B}(S_i, \epsilon)$
- Kolmogorov Entropy $H_\epsilon(K) := \log_2 N_\epsilon(K)$

- $\delta_n(K) = \inf\{\epsilon : \; H_\epsilon(K) \leq n\}$

Description of Optimal Encoding: Kolmo

- Given $\epsilon > 0$
- Minimal ϵ cover: $K \subset \cup_{i=1}^{N_\epsilon} \mathcal{B}(x_i, \epsilon)$
- $\delta_n(K) = \inf\{\epsilon : \ H_\epsilon(K) \leq n$
- Kolmogorov entropy of K gives our benchmark
- Usually not practical encoder

The Issues

1. The metric: least squares
2. The classes
3. Determine Entropy of Classes
4. Build near optimal Encoders/Decoders

Wavelet Encoding

- Tree approximation can be turned into an encoder

Wavelet Encoding

- Tree approximation can be turned into an encoder
- Tree with n nodes can be encoded with n bits

Wavelet Encoding

- Tree approximation can be turned into an encoder
- Tree with n nodes can be encoded with n bits
- Cohen-Dahmen-Daubechies-DeVore (Cohen-Daubechies-Gulleryuz-Orchard) This encoder is optimal on all Besov classes compactly embedded into L_2

Wavelet Encoding

- Tree approximation can be turned into an encoder
- Tree with n nodes can be encoded with n bits
- Cohen-Dahmen-Daubechies-DeVore (Cohen-Daubechies-Gulleryuz-Orchard) This encoder is optimal on all Besov classes compactly embedded into L_2
- EZW, Said-Pearlman,